Accredited

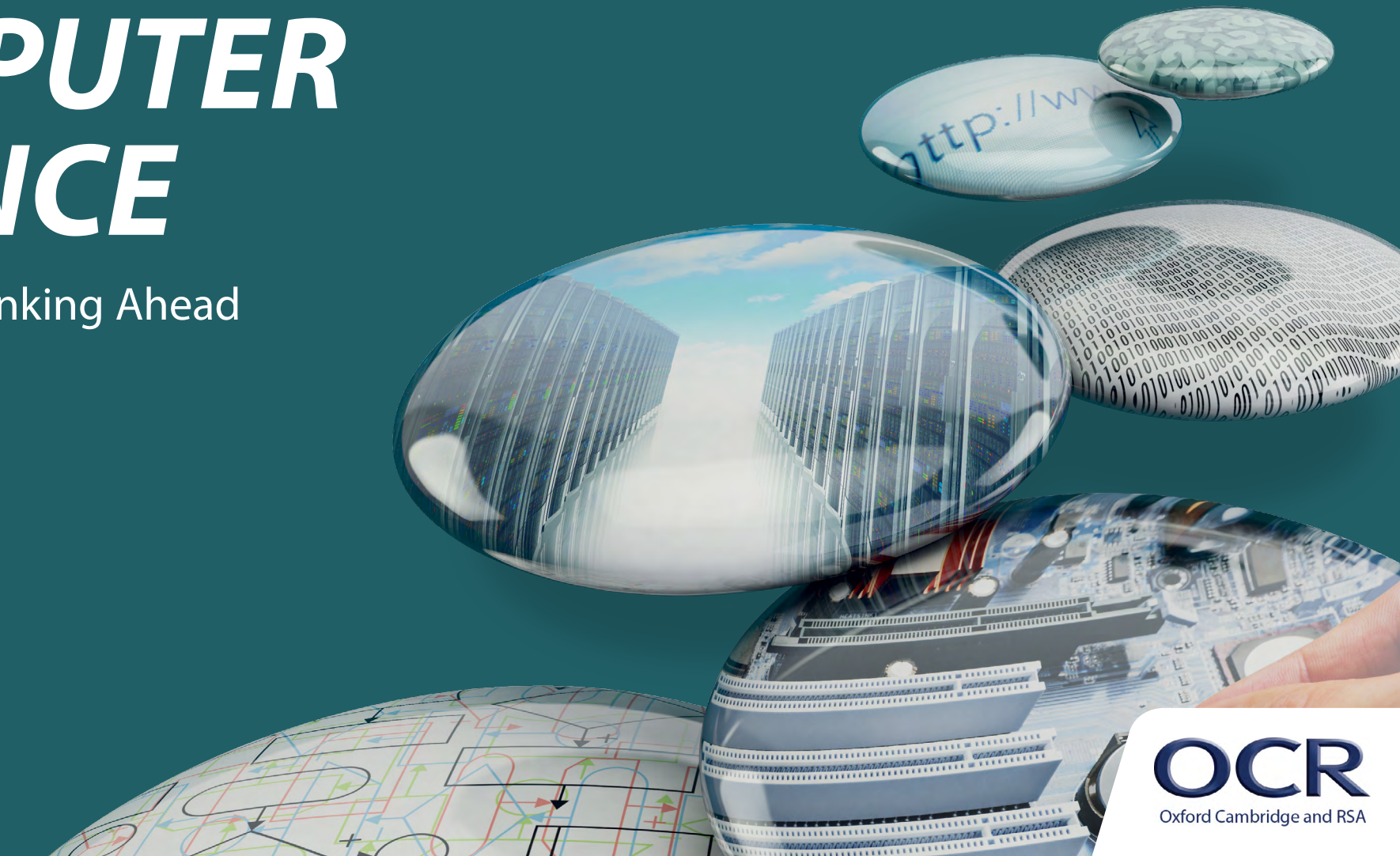
A LEVEL
Delivery Guide

H446

COMPUTER SCIENCE

Theme: 2.1.2 Thinking Ahead

June 2015



OCR
Oxford Cambridge and RSA

We will inform centres about any changes to the specification. We will also publish changes on our website. The latest version of our specification will always be the one on our website (www.ocr.org.uk) and this may differ from printed versions.

Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee.
Registered in England. Registered company number 3484466.

Registered office: 1 Hills Road
Cambridge
CB1 2EU

OCR is an exempt charity.

CONTENTS

Introduction

Delivery guides are designed to represent a body of knowledge about teaching a particular topic and contain:

- Content: a clear outline of the content covered by the delivery guide;
- Thinking Conceptually: expert guidance on the key concepts involved, common difficulties students may have, approaches to teaching that can help students understand these concepts and how this topic links conceptually to other areas of the subject;
- Thinking Contextually: a range of suggested teaching activities using a variety of themes so that different activities can be selected that best suit particular classes, learning styles or teaching approaches.

If you have any feedback on this Delivery Guide or suggestions for other resources you would like OCR to develop, please email resources.feedback@ocr.org.uk.

KEY

Click to view associated resources within this document.

Click here — Click to view external resources

AS Level only — AS Level content only

Curriculum Content

a) Identify the inputs and outputs for a given situation.

b) Determine the preconditions for devising a solution to a problem.

c) The nature, benefits and drawbacks of caching.

d) The need for reusable program components.

Curriculum Content

Activities	Resources
In this unit students will be required to identify the inputs and outputs for a given situation; this could relate to an IT system that a business uses or a programming/software solution. Once these are identified, a flow chart can be created to show in diagrammatical form the inputs and outputs of the system. Students should be encouraged to create a flowchart identifying and representing the various elements of a system. http://www.rff.com/flowchart_shapes.htm	Click here
Students will learn about the impact of caching in relation to a programming solution or IT system. They could use their knowledge to create a user guide or instructional video which explains the nature, benefits and drawbacks of caching. Scan Disk provides a good starting discussion on this topic: https://www.youtube.com/watch?v=L8gzBLTTBYU Students' understanding might also be supported by developing their overall technical understanding of caching, how it works and how it is used. http://computer.howstuffworks.com/cache.htm	Click here Click here
The need for reusable program components is important from a business point of view as well as a tool to support future developments. http://en.wikipedia.org/wiki/Code_reuse Students could be asked to investigate one particular method of software reuse using this site as a beginning point for the theory: http://epf.eclipse.org/wikis/openup/core.tech.common.extend_supp/guidances/guidelines/software_reuse_B6B04C26.html Further reading is provided in this pdf: http://gameproductionuu.files.wordpress.com/2014/01/11-reuse.pdf Students could consider what elements of their own programs are reusable; this may begin as a simple definition or function. They could annotate printouts of their programs and discuss these elements.	Click here Click here Click here
Determining the preconditions for devising a solution to a problem could begin with looking at a scenario within the school or institution where the students are learning; perhaps replacing or upgrading a computer suite, or introducing mobile devices. Students could list and discuss these, giving consideration to how they will influence a project. For example, if a business was creating a website and the company already employed two programmers fluent in HTML5, this would impact on the solution in that these employees would be used, and the website would probably be coded in HTML5. It is important for students to consider any tools, skills, strengths, or weaknesses they may have before pursuing a project solution.	

Thinking Conceptually

In this unit the content covers the requirement for identifying the inputs and outputs for a given situation. This is usually presented in diagram form as parts in a flow chart showing the inputs, outputs, processes and decisions. Logically ordering the sections of a solution into these parts supports the development of a solution.

before devising a system solution, students need to be able to consider the current systems that are being used and how they are used across the business. This section focuses on problem solving and considering project planning. It is important to consider how the current solution will affect the new solution. Within these considerations are included the skill level that is required to create the solution, do the programmers have the skills or knowledge to produce the solution: if it is a web solution is there a skills base for using JavaScript (which leads to discussions around which language should be used)? Will the new system complement the users'/customers' needs?

A current system's reusability may be an outcome of the need for reusable program components. For example, standard interfaces allow for interchangeable, reusable components. Consider a simple program icon which is then used throughout a suite of programs: the 'Save' icon within the ribbon of a word processor, including its functionality, is reused throughout other software applications such as spreadsheets or databases. A business that systematically reuses these components can combine them as the building blocks of other systems. Sometimes this approach is systematic and a planned part of the development process.

The majority of systems store data and this will be an area that students will be familiar with. Caching in traditional terms concerns storing data or instructions in high speed accessible memory. This results in faster response times and overall better device performance. Students will be required to look at how modern applications run across a number of devices, platforms and servers. This data is often cached across a number of hardware servers. Software takes this shared cache of data and combines it to create one single viewable image, even though it is stored across several machines. Traditionally, as more devices are added to a system stability can slow down and this creates bottlenecks. Synchronising the data across a number of servers or running applications on specific servers can reduce the overall dependency on a specific server, thus reducing bottlenecks.

Caching across servers enables users to distribute applications to multiple physical locations, multiple departments within an organisation, and multiple businesses across the Internet. Plus, it permits reuse of existing code within an organisation and, more importantly, fosters collaboration among different business units.

However, students need to consider the related concerns of caching web application data, such as storing data of a sensitive or personal nature.

Thinking Contextually

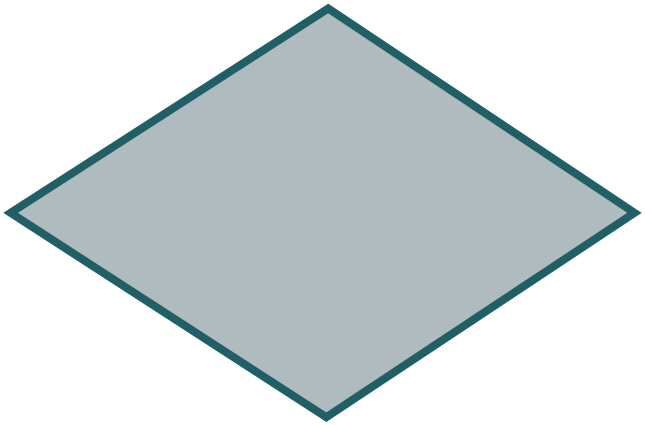
Activities	Resources
Activity 1: inputs and outputs Programs will have a number of inputs and outputs that form some element of the final outcome of the coding solution. These inputs are combined with processes to create outcomes. Consider a simple game of hangman to find a six letter word. 1) List all the Inputs 2) List the outputs. Now consider the processes and decisions that take place when a letter is selected or entered. The letter is either correct or incorrect – this can be considered as a process, the response that is fed back to the player is an output. The three elements, inputs, outputs and processes, are represented by the following symbols. • An input or output is a parallelogram. • A process is a rectangle. • A decision is a diamond. Show the symbols in Learner Resource 1 to the students. Ask each learner to: 1) Create a flowchart diagram showing the inputs, outputs, processes and decisions for the hangman game. 2) Compare with another student's answer and discuss the parts – make improvements to their diagram.	Learner Resource 1
Activity 2: reusable program components 1) Cut up the cards in Learner Resource 2 and share them between students/groups of students. Find definitions for the various forms of reusable program components. 2) Get students to order or rank them in terms of which components are most important or useful. Ask students to discuss and justify their choices. 3) The teacher or a student reads out an example of a reusable component and students hold up the matching type card. See Learner Resource 3.	Learner Resource 2 Learner Resource 3

Thinking Contextually

Activities	Resources
Activity 3 You have been approached by a business who has asked you to make a document regarding the nature of caching, identifying and evaluating the benefits and drawbacks and giving examples. This could be a video, animation, presentation or a report. Areas of focus: • Introduce what caching is and the requirement for it. • Examples of where caching is used. • Briefly describe how caching works. • Discuss the benefits of caching. • Discuss the drawbacks. • Focus on, for example, scalability, maintenance issues, security and synching issues.	

Learner Resource 1 Activity 1 – process element symbols

See page 8



Learner Resource 2 Activity 2 – reusable program components

See page 8

Function/definition of reuse	Component reuse – subsystem reusable in another application	Standard programming language reuse across all projects
Using an entire component across a suite of programs	A reusable software module or library	External reuse – reselling a component to a third party
Planned reuse where a company designs elements that will be useable again	Generator based reuse	Program generator

Learner Resource 3 Activity 2 – component cards

See page 8

Import the Random Module	A reusable software module or library
A menu toolbar	Using an entire component across a suite of programs
Using Java across all program solutions	Standard programming language reuse across all projects
API, for example, Twitter	External reuse – reselling a component to a third party
Company or project ethos building in reusability	Planned reuse where a company designs elements that will be useable again
The reuse of a standard algorithm or pattern of instructions	Generator based reuse
A program that allows the user to easily create their own solution with minimal or sometimes no programming	Program generator


We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here:
http://www.ocr.org.uk/qualifications/expression-of-interest/

OCR customer contact centre
General qualifications
Telephone 01223 553998
Facsimile 01223 552627
Email general.qualifications@ocr.org.uk